

**THIS PAGE IS INSERTED BY OIPE SCANNING
AND IS NOT PART OF THE OFFICIAL RECORD**

Best Available Images

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images may include (but are not limited to):

BLACK BORDERS

TEXT CUT OFF AT TOP, BOTTOM OR SIDES

FADED TEXT

BLURRY OR ILLEGIBLE TEXT

SKEWED/SLANTED IMAGES

COLORED PHOTOS HAVE BEEN RENDERED INTO BLACK AND WHITE

VERY DARK BLACK AND WHITE PHOTOS

UNDECIPHERABLE GRAY SCALE DOCUMENTS

**IMAGES ARE THE BEST AVAILABLE
COPY. AS RESCANNING *WILL NOT*
CORRECT IMAGES, PLEASE DO NOT
REPORT THE IMAGES TO THE
PROBLEM IMAGE BOX.**

(19) World Intellectual Property Organization
International Bureau



(43) International Publication Date
31 May 2001 (31.05.2001)

PCT

(10) International Publication Number
WO 01/39024 A2

(51) International Patent Classification⁷: G06F 17/00

(21) International Application Number: PCT/US00/32250

(22) International Filing Date:
22 November 2000 (22.11.2000)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
60/166,953 22 November 1999 (22.11.1999) US
09/575,917 23 May 2000 (23.05.2000) US

(71) Applicant: AVENUE A, INC. [US/US]; 9th Floor, The Smith Tower, 506 Second Avenue, Seattle, WA 98104 (US).

(72) Inventors: LIPSKY, Scott, Eric; 2211 32nd Avenue South, Seattle, WA 98144 (US). BARTLETT, James,

Patrick; 10822 SE 173rd Street, Renton, WA 98055 (US). BECK, Steve; 20019 27th Drive SE, Bothell, WA 98012 (US). SCHUSTER, Gary, Richard; 2742 215th Place SW, Brier, WA 98036 (US). SPINARSKI, John, Bryan; 18316 NE 25th Street, Redmond, WA 98052 (US).

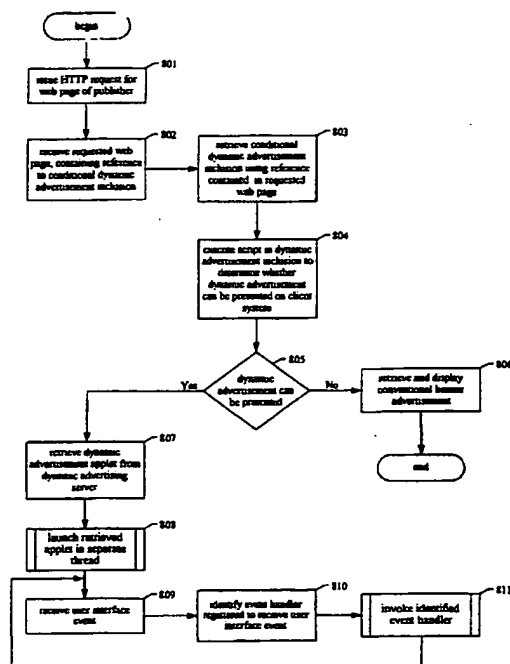
(74) Agents: LAWRENZ, Steven, D. et al.; Perkins Coie LLP, P.O. Box 1247, Seattle, WA 98111-1247 (US).

(81) Designated States (*national*): AE, AG, AL, AM, AT, AU, AZ, BA, BB, BG, BR, BY, BZ, CA, CH, CN, CR, CU, CZ, DE, DK, DM, DZ, EE, ES, FI, GB, GD, GE, GH, GM, HR, HU, ID, IL, IN, IS, JP, KE, KG, KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, MA, MD, MG, MK, MN, MW, MX, MZ, NO, NZ, PL, PT, RO, RU, SD, SE, SG, SI, SK, SL, TJ, TM, TR, TT, TZ, UA, UG, UZ, VN, YU, ZA, ZW.

(84) Designated States (*regional*): ARIPO patent (GH, GM, KE, LS, MW, MZ, SD, SL, SZ, TZ, UG, ZW), Eurasian patent (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European patent (AT, BE, CH, CY, DE, DK, ES, FI, FR, GB, GR, IE,

[Continued on next page]

(54) Title: DYNAMIC INTERNET ADVERTISING



(57) Abstract: A facility for presenting dynamic advertising in a client computer system is described. The facility receives a specified sequence for advertising messages that identifies advertising messages by reference. The facility retrieves advertising messages identified in the specified sequence and displays them in accordance with the specified sequence. The facility further collects information describing interaction by a user of the client computer system with the displayed advertising messages. The facility transmits this collected information to a separate information processing computer system.



IT, LU, MC, NL, PT, SE, TR), OAPI patent (BF, BJ, CF, CG, CI, CM, GA, GN, GW, ML, MR, NE, SN, TD, TG).

For two-letter codes and other abbreviations, refer to the "Guidance Notes on Codes and Abbreviations" appearing at the beginning of each regular issue of the PCT Gazette.

Published:

- *Without international search report and to be republished upon receipt of that report.*

DYNAMIC INTERNET ADVERTISING

TECHNICAL FIELD

The present invention is directed to Internet advertising techniques, and, more particularly, to client-side advertising mechanisms.

5 BACKGROUND

The World Wide Web ("the Web") is a medium for making content available to requesting users, also known as "publishing" content. A Web publisher may provide content of various types, including visual Web pages having textual and static visual contents, as well as such other
10 content such as animated images and audio and video sequences. The relatively low cost of publishing content on the Web enables many organizations and individuals to act as Web publishers.

To access content made available on the Web, a user typically directs a Web client program, or "browser," executing on his or
15 her computer system to obtain and display a particular unit of content. Units of content are each identified by an address, also called a Uniform Resource Locator, or URL. The user may direct the browser to obtain and display a unit of content by directly entering the URL for the unit of content, or by selecting a link or bookmark with which the URL is
20 associated.

Internet advertising is a practice in which a Web publisher adds an advertising message or other content provided by a third party advertiser to the content provided by the Web publisher in exchange for payment or other consideration from the advertiser. For example, a Web-
25 based travel agency may pay the publisher of a Web site directed to tips for travelers to add a banner advertisement for the Web-based travel agency to some of the Web pages comprising the Web site. Such a

banner advertisement typically presents visual information promoting the advertiser that takes up a portion of the area of the Web page. Some banner advertisements further constitute a link to a Web site associated with the advertiser, so that the user may click anywhere within the banner advertisement to display, or "click through to," that Web site.

As computer use, and particularly the use of the Web, becomes more and more prevalent, the volume of Internet advertising presented grows larger and larger. As this volume continues to increase, the need to differentiate advertising messages from those of competitors, the need to maximize the number of advertising messages presented in a given amount of advertising space, and the need to collect information about the effectiveness of advertising increases significantly.

Accordingly, a facility for presenting dynamic Internet advertising that monitors, reports on, and responds to user behavior would have significant utility.

BRIEF DESCRIPTION OF THE DRAWINGS

Figure 1 is a high-level block diagram showing the environment in which the facility preferably operates.

Figure 2 is a screenshot diagram showing the display of a dynamic advertisement displayed by the facility.

Figure 3 is a display diagram showing a first stage of the visual transition.

Figure 4 is a display diagram showing a second stage of the visual transition.

Figure 5 is a display diagram showing complete replacement of the first advertising message by the second advertising message.

Figure 6 is a table diagram showing the definition of subregions within the dynamic advertising region.

Figure 7 is a display diagram showing the display of a Web page in response to the user clicking within a subregion of the first advertising message.

Figure 8 is a flow diagram showing the steps preferably performed by the facility when the user navigates to a publisher Web page on which space has been purchased for dynamic advertising.

Figure 9 is a flow diagram showing the steps preferably performed when the dynamic advertising applet is launched in a separate thread.

Figure 10 is a flow diagram showing the steps preferably performed in an event handler registered by the facility to process user input events occurring in conjunction with the dynamic advertising region.

DETAILED DESCRIPTION

A software facility for presenting dynamic Internet advertising that monitors, reports on, and responds to user behavior is provided. The facility uses a client-side component, such as a Java applet executing on the user's computer system in conjunction with the Web browser being used by the user. The client-side component composes advertising messages from multi-media elements including graphics, text, audio, video, and various data entry elements and presents them on the user's computer system based upon the contents of a script retrieved from a dynamic advertising server by the client-side component. The script received by the client-side component specifies such details of the client-side component's operation as the sequence of advertising messages, transitions between advertising messages, mappings from specific subregions defined within the dynamic advertising region to different click-through targets and text superimposed over the advertising messages. By sending the client-side component an altered script, the dynamic advertising server can quickly adjust any aspects of the behavior

of the client-side component, while consuming just the small amount of network/data bandwidth required by the script.

The client-side component is preferably able to autonomously modify or replace such advertising messages, irrespective of whether a particular action is taken by the user. The client-side component preferably also collects and transmits to a central server information about the user's behavior, such as where the user has moved the mouse pointer, the times at which the mouse pointer enters and exits that area, whether the user has clicked on any user interface controls, when the advertising message was first and last viewable in the browser, etc. In accordance with additional aspects of the facility, data received from each client-side component is preferably analyzed by the central server. In some embodiments, the central server sends new instructions based upon the analysis back to the client-side component. The client-side component provides advertising messages that are proactive, dynamic, and adapted to the behavior of the user.

Because it controls the display of advertising messages, the client-side component enables advertising messages for multiple advertisers to be displayed sequentially or simultaneously in the same ad space. In some embodiments, the facility provides real-time or occasional integration with advertiser databases in order to deliver advertising messages based upon information retrieved by the client-side component from advertiser databases, such as current pricing and availability information.

Figure 1 is a high-level block diagram showing the environment in which the facility preferably operates. The diagram shows a client computer system 110. An Internet user preferably uses the client computer system to connect, via the Internet 100, to an Internet publisher computer system, such as Internet publisher computer systems 130 and 140, to retrieve and display a Web page.

In cases where an Internet advertiser, through the Internet advertising service, has purchased advertising space on the Web page provided to the Internet user computer system by the Internet publisher computer system, the Web page contains a reference that causes the client
5 computer system to retrieve from a dynamic advertising server 120 the client side dynamic advertising component and execute it. When executed, the client-side component downloads the appropriate dynamic advertising script from the dynamic advertising server 120, which specifies the sequence of advertising messages to display. The client-side
10 component retrieves and presents these advertising messages within the publisher Web page. The script specifies the server from which the advertising messages are to be downloaded. Different advertising messages may be downloaded from different servers, such as Internet content distributor computer systems 170 and 180. While presenting
15 these advertising messages, the client-side component monitors and stores in a log user interface events that occur with respect to the displayed sequence of advertising messages. The client-side component periodically transmits these events to the dynamic advertising server.

When the user clicks within the dynamic advertising
20 message, the client-side component redirects the Internet user computer system to retrieve a Web page from the appropriate Internet advertiser computer system, such as Internet advertiser computer system 150 or 160. In visiting the Internet advertiser's Web site, the Internet user may traverse several pages, and may take such actions as purchasing an item or
25 bidding in an auction. Revenue from such actions typically finances, and is often the motivation for, the Internet advertiser's Internet advertising.

The client computer system 110 preferably includes one or more central processing units (CPUs) 111 for executing computer programs such as the facility, a computer memory 112 for storing
30 programs and data, a persistent storage device 113, such as a hard drive, and a computer-readable media drive 114, such as a CD-ROM drive, for

reading programs and data stored on a computer-readable medium. The dynamic advertising sever 120 preferably includes components similar to those described in the client computer system 110.

While preferred embodiments are described in terms of the environment described above, those skilled in the art will appreciate that the facility may be implemented in a variety of other environments, including a single, monolithic computer system, as well as various other combinations of computer systems or similar devices.

In order to more fully describe the implementation of the facility, its operation is described in conjunction with an example.

Figure 2 is a screenshot diagram showing the display of a dynamic advertisement displayed by the facility. The diagram shows a browser window 200, which includes a client area 210 in which is displayed a sample publisher Web page. In addition to other contents, the sample Web page contains a dynamic advertising region 220 managed by the client-side component of the facility. In the dynamic advertising region, the client-side component has displayed a first advertising message 221 for Advertiser A. It can be seen that the first advertisement includes a rectangular image, over which the client-side component has superimposed the following text 231: "Dynamic Text Here". The client-side component preferably displays this first advertisement for a period of time specified by the script, such as 10 seconds. During this time, the client-side component tracks any user interface events received in the browser that relate to the dynamic advertising regions. The client-side component stores these events in a log for later transmission to the dynamic advertising server.

Figures 3-5 show a visual transition performed by the facility from the display of the first advertising message to the display of the second advertising message. Figure 3 is a display diagram showing a first stage of the visual transition. It can be seen in this diagram that, in dynamic advertising region 320, the client-side component shows a

portion of a second advertising message 322 that replaces the left side of the first advertising message 321.

Figure 4 is a display diagram showing a second stage of the visual transition. It can be seen in this diagram that, in dynamic advertising region 420, the client-side component shows a larger portion of a second advertising message 422 that replaces the left side of the first advertising message 421.

Figure 5 is a display diagram showing complete replacement of the first advertising message by the second advertising message. The facility preferably displays the second advertising message 522 for a period of time specified by the script, such as 10 seconds, continuing to track and record user interface events. After displaying the second advertising message, the facility again displays the first advertising message.

Were the user to perform a user interface event with respect to the dynamic advertising region, such as moving the mouse from outside the dynamic advertising region into the dynamic advertising region while the client-side component is displaying a visual transition from the first advertising message to the second advertising message, such as is shown in Figures 3 and 4, the client-side component preferably reverts to the display of the first advertising message shown in Figure 2 to permit the user to further review and interact with the first advertising message.

Figure 6 is a table diagram showing the definition of subregions within the dynamic advertising region. Table 600 contains two copies of the first advertising message, 610 and 620, each showing one of two subregions defined with the first advertising message by the dynamic advertising script. Rectangle 612 in the first copy of the first advertising message is the border of a first subregion defined in the first advertising message. Rectangle 622 in the second copy of the first advertising message defines the second region within the first advertising message. The table further shows that, when the user clicks within

subregion 612, the client-side component causes the browser to navigate to the following URL 613: "http://www.advertiser.com". The table also shows that, when the user clicks within region 622, the client-side component causes the browser to navigate to URL 623:
5 "http://www.avenue.com".

For example, Figure 7 is a display diagram showing the display of the Web page at URL 623 in response to the user clicking within subregion 622 using the mouse pointer 621.

Those operating the dynamic advertising service preferably
10 prepare a number of different scripts, or dynamic advertising specifications, each specifying a particular instance of dynamic advertising. A script preferably identifies the advertising messages that are to be presented, such as image files, preferably by specifying a URL from which each may be retrieved; the order in which the advertising
15 messages are to be displayed and details of the visual transition to be performed in switching from displaying one advertising message to the next; definitions of regions within one or more of the advertising messages, and attributes of the regions such as click-through destinations to which to navigate if the user clicks within the region, and text to be
20 displayed within the region. Scripts are preferably small in size, allowing the performance of a particular client-side component to be redefined merely by sending a revised script to the client-side component. Indeed, such a revised script may identify the same advertising messages as the original script, enabling the browser to obtain those creatives from its
25 cache rather than re-retrieving them.

Scripts are preferably prepared in the script language described below in Table 1, though those skilled in the art will appreciate that a variety of other formats could be gainfully used to represent a dynamic advertising specification.

30

- **DynamicAdvertisingScript** xmlns="x-schema:client-side componentschema.xml"> - This tag denotes the beginning of the script as well as what schema (rules) to follow when parsing the script. Each script can contain the following elements
 - **<HeartBeat TimeValue=10>** - Heartbeat is the interval at which the client should report statistics back to the server. TimeValue is the interval in seconds between sending statistics to the server
 - **<TransitionSpeed TimeValue=4>** - Denotes the speed of transitioning from one image to another. the lower the number the faster the transition - 0 means immediate. This value is the number of milliseconds between updating the coordinates of the image.
 - **<GraphicText >** - Denotes text to display on the Applet, there can be only one graphic text object at the root of the script, and zero-to-many graphic text objects per image elements. A graphic text element can contain the following elements
 - **<DisplayText>** - Text to display, there can be many of these per Graphic text element, each one will display one line below the prior one
 - **** - This is the font to use for the text.
 - **<BGColor Red="255" Green="255" Blue="255"/>** - Background color for the text
 - **<FGColor Red="0" Green="0" Blue="0"/>** - Foreground color for the text
 - **<Rectangle RectX="200" RectY="0" RectW="150" RectH="50"/>** - Bounding rectangle to draw the text in, if there is no bounding rectangle then the entire window space available will be used.
 - **<Image Location="mtvnews.gif">** - Denotes the definition of one Image, must have a location property that tells the client where to load the image from. The location can be a relative URL based on the code base of the applet, or a fully qualified URL such as HTTP://www.advertiser.com/webpage. An image can contain the following elements
 - **<DefaultClick Location="HTTP://www.advertiser.com/default"/>** - This is the URL that will be browsed to if a user clicks on the client but is not in a defined region. One DefaultClick element is allowed per Image.
 - **<DisplayTime>** - How long to Display each image. This value is represented in seconds.
 - **<Region>** - Defines a region for the current image, a region contains the following elements
 - **<Rectangle RectX="0" RectY="0" RectW="234" RectH="30"/>** - Defines the upper left corner, width, and height for the region, in pixels.
 - **<URL Location="HTTP://www.advertiser.com/adtarget"/>** - Defines the URL to browse to when the region is clicked on.

Table 1

An example script expressed in the script language shown in

Table 1 above is shown below in Table 2.

```

1. <?xml version="1.0" ?>
2. <Client-side componentScript xmlns="x-schema:client-side componentschema.xml">
3.   <HeartBeat TimeValue="10"/>
4.   <TransitionSpeed TimeValue="10"/>
5.   <Image Location="AdImage1_export.gif">
6.     <DefaultClick Location="http://www.avenuea.com"/>
7.     <Region>
8.       <Rectangle RectX="400" RectY="0" RectW="168" RectH="60"/>
9.       <URL Location="http://www.advertiser.com"/>
10.    </Region>
11.    <GraphicText>
12.      <DisplayText>Dynamic Text Here</DisplayText>
13.      <Font FontName="times" Face="BoldItalic" PointSize="12"/>
14.      <FGColor Red="0" Green="0" Blue="0"/>
15.      <Rectangle RectX="130" RectY="0" RectW="200" RectH="30"/>
16.    </GraphicText>
17.    <DisplayTime TimeValue="2"/>
18.  </Image>
19.  <Image Location="AdImage2_export.gif">
20.    <DefaultClick Location="http://www.avenuea.com"/>
21.    <Region>
22.      <Rectangle RectX="130" RectY="0" RectW="200" RectH="30"/>
23.      <URL Location="http://www.advertiser.com"/>
24.    </Region>
25.    <GraphicText>
26.      <DisplayText>Different Dynamic Text Here</DisplayText>
27.      <Font FontName="times" Face="BoldItalic" PointSize="12"/>

```

5

```

28.         <FGColor Red="0" Green="0" Blue="0"/>
29.         <Rectangle RectX="130" RectY="0" RectW="200" RectH="30"/>
30.       </GraphicText>
31.       <DisplayTime TimeValue="2"/>
32.     </Image>
33. </Client-side componentScript>

```

Table 2

10

Line 3 specifies that the client-side component should send feedback to the dynamic advertising server every ten seconds. Line 4 specifies that, when transitioning from one advertising message to the next, ten milliseconds should elapse between updating the coordinates of the advertising message. By specifying various values for the TimeValue

15

property of the TransitionTime tag, the script author can specify how long it takes to transition from one advertising message to the next, or, by specifying zero, can specify that transitions are to be instantaneous. Lines 5-18 specify details of a first advertising message, while lines 19-30 specify details of a second advertising message. With respect to the first

20

advertising message, line 5 specifies the URL of a graphics file containing the image for the advertising message. Line 6 contains the click-through URL for positions within the dynamic advertising region that are not within any defined sub-regions. Lines 7-10 define one sub-region within the dynamic advertising region. Line 8 specifies its boundaries, while 9

25

specifies its click-through URL. Lines 11-16 specify the display of the text "Dynamic Text Here" in the first advertising message. Lines 19-30 have similar contents with respect to the second advertising message. Thus, it can be seen that different regions may be defined within an advertising message for different purposes.

30

Once a script such as the script shown in Table 2 has been generated, the facility preferably checks the script for compliance with an XML schema defining the proper syntax for a dynamic advertising script. One such preferred schema is shown below in Table 3.

35

```

1. <Schema xmlns="urn:schemas-microsoft-com:xml-data"
2.   xmlns:dt="urn:schemas-microsoft-com:datatypes">

```

3. <AttributeType name="TimeValue" dt:type="int"/>
4. <Element Type name="DisplayTime" content="empty">
5. <attribute type="TimeValue" required="yes"/>
6. </Element Type>
7. <AttributeType name="Red" dt:type="int"/>
8. <AttributeType name="Green" dt:type="int"/>
9. <AttributeType name="Blue" dt:type="int"/>
10. <Element Type name="BGColor" content="empty">
11. <attribute type="Red" required="yes"/>
12. <attribute type="Green" required="yes"/>
13. <attribute type="Blue" required="yes"/>
14. </Element Type>
15. <Element Type name="FGColor" content="eltOnly">
16. <attribute type="Red" required="yes"/>
17. <attribute type="Green" required="yes"/>
18. <attribute type="Blue" required="yes"/>
19. </Element Type>
20. <AttributeType name="FontName" dt:type="string"/>
21. <AttributeType name="Face" dt:type="enumeration" dt:values="Bold Normal Italic BoldItalic"/>
22. <AttributeType name="PointSize" dt:type="int"/>
23. <Element Type name="Font" content="empty">
24. <attribute type="FontName" required="yes"/>
25. <attribute type="Face" required="yes"/>
26. <attribute type="PointSize" required="yes"/>
27. </Element Type>
28. <Element Type name="DisplayText" content="textOnly"/>
29. <AttributeType name="RectX" dt:type="int"/>
30. <AttributeType name="RectY" dt:type="int"/>
31. <AttributeType name="RectH" dt:type="int"/>
32. <AttributeType name="RectW" dt:type="int"/>
33. <Element Type name="Rectangle" content="empty">
34. <attribute type="RectX" required="yes"/>
35. <attribute type="RectY" required="yes"/>
36. <attribute type="RectH" required="yes"/>
37. <attribute type="RectW" required="yes"/>
38. </Element Type>
39. <Element Type name="GraphicText" content="eltOnly" order="many">
40. <element type="DisplayTime" minOccurs="0" maxOccurs="1"/>
41. <element type="DisplayText" minOccurs="1" maxOccurs="1"/>
42. <element type="Font" minOccurs="0" maxOccurs="1"/>
43. <element type="BGColor" minOccurs="0" maxOccurs="1"/>
44. <element type="FGColor" minOccurs="0" maxOccurs="1"/>
45. <element type="Rectangle" minOccurs="0" maxOccurs="1"/>
46. </Element Type>
47. <AttributeType name="Location" dt:type="string"/>
48. <Element Type name="URL" content="empty">
49. <attribute type="Location" required="yes"/>
50. </Element Type>
51. <Element Type name="DefaultClick" content="empty">
52. <attribute type="Location" required="yes"/>
53. </Element Type>
54. <Element Type name="Region" content="eltOnly" order="many">
55. <element type="Rectangle" minOccurs="1" maxOccurs="1"/>
56. <element type="URL" minOccurs="1" maxOccurs="1"/>
57. </Element Type>
58. <Element Type name="Image" content="eltOnly" order="many">
59. <attribute type="Location" required="yes"/>
60. <element type="DefaultClick" minOccurs="1" maxOccurs="1"/>
61. <element type="Region" minOccurs="0" maxOccurs="1"/>
62. <element type="DisplayTime" minOccurs="1" maxOccurs="1"/>
63. <element type="GraphicText" minOccurs="0" maxOccurs="1"/>
64. </Element Type>
65. <Element Type name="HeartBeat" content="empty">
66. <attribute type="TimeValue" required="yes"/>
67. </Element Type>
68. <Element Type name="TransitionSpeed" content="empty">
69. <attribute type="TimeValue" required="yes"/>
70. </Element Type>
71. <Element Type name="Client-side componentScript" content="eltOnly" order="many">
72. <element type="DefaultClick" minOccurs="1" maxOccurs="1"/>
73. <element type="HeartBeat" minOccurs="1" maxOccurs="1"/>
74. <element type="TransitionSpeed" minOccurs="1" maxOccurs="1"/>
75. <element type="Image" minOccurs="1" maxOccurs="1"/>
76. <element type="GraphicText" minOccurs="0" maxOccurs="1"/>

```

77.         <element type="BGColor" minOccurs="0" maxOccurs="1"/>
78.         <element type="FGColor" minOccurs="0" maxOccurs="1"/>
79.     </ElementType>
80. </Schema>

```

5

Table 3

After insuring that the dynamic advertising script conforms to the XML schema for advertising scripts, the facility preferably
 10 compresses the dynamic advertising script, employing a numerical substitution scheme such as the one detailed below in Table 4.

Element/Attribute Name	Binary Representation of Value
DynamicAdvertisingScript	"1"
HeartBeat	"2" + <1 byte for # of seconds>
Image	"3" + <2 bytes for length> + <Character string for location>
Region	"4"
Rectangle	"5" + <2 bytes for X> + <2 bytes for Y> + <2 bytes for H> + <2 bytes for W>
URL	"6" + <2 bytes for length> + <Character string for location>
DisplayTime	"7" + <1 byte for # of seconds>
TransitionSpeed	"8" + <1 byte for # of milliseconds>
DefaultClick	"9" + <2 bytes for length> + <Character string for location>
GraphicText	"10"
DisplayText	"11" + <2 bytes for length> + <Character string to display>
Font	"12" + <2 bytes for length> + + <1 byte for face> + <1 byte for point size>
FGColor	"13" + <2 bytes for red> + <2 bytes for blue> + <2 bytes for green>
BGColor	"14" + <2 bytes for red> + <2 bytes for blue> + <2 bytes for green>

Each element is delimited by the negative value of its binary representation to signify the "end" of that element, for example DefaultClick element would look as follows

15

"9" + <2 bytes for length> + <Character string for location>-9

Table 4

Table 5 below shows the compressed, or "binary" form of
 20 the sample script shown in Table 2 generated by applying the substitution scheme described in Table 4.

	1.	01 02 0A FE 08 0A F8 03-00 13 41 64 49 6D 61 67AdImag
	2.	65 31 5F 65 78 70 6F 72-74 2E 67 69 66 09 00 16	el_export.gif...
	3.	68 74 74 70 3A 2F 2F 77-77 77 2E 61 76 65 6E 75	http://www.avenue
5	4.	65 61 2E 63 6F 6D F7 04-05 01 90 00 00 00 A8 00	ea.com.....
	5.	3C FB 06 00 1A 68 74 74-70 3A 2F 2F 77 77 77 2E	<....http://www.
	6.	61 64 76 65 72 74 69 73-65 72 61 2E 63 6F 6D FA	advertiser.com.
	7.	FC 0A 0B 00 11 44 79 6E-61 6D 69 63 20 54 65 78Dynamic Tex
	8.	74 20 48 65 72 65 F5 0C-00 05 74 69 6D 65 73 03	t Here.....times:
10	9.	0C F4 0D 00 00 00 00 00-00 F3 05 00 82 00 00 00
	10.	C8 00 1E FB F6 07 02 F9-FD 03 00 13 41 64 49 6DAdIm
	11.	61 67 65 32 5F 65 78 70-6F 72 74 2E 67 69 66 09	age2_export.gif.
	12.	00 16 68 74 74 70 3A 2F-2F 77 77 77 2E 61 76 65	..http://www.ave
	13.	6E 75 65 61 2E 63 6F 6D-F7 04 05 00 82 00 00 00	nuea.com.....
15	14.	C8 00 1E FB 06 00 1A 68-74 74 70 3A 2F 2F 77 77http://ww
	15.	77 2E 61 64 76 65 72 74-69 73 65 72 62 2E 63 6F	w.advertiserb.cc
	16.	6D FA FC 0A 0B 00 1B 44-69 66 66 65 72 65 6E 74	m.....Different
	17.	20 44 79 6E 61 6D 69 63-20 54 65 78 74 20 48 65	Dynamic Text He
	18.	72 65 F5 0C 00 05 74 69-6D 65 73 03 0C F4 0D 00	re.....times.....
20	19.	00 00 00 00 00 F3 05 00-82 00 00 00 C8 00 1E FB
	20.	F6 07 02 F9 FD FF 80 00-AA 81 CD 00 40 EB CE E8&...

Table 5

Figures 8-10 are flow diagrams showing steps preferably performed in the client computer system to display dynamic advertisements using the facility. Figure 8 is a flow diagram showing the steps preferably performed by the facility when the user navigates to a publisher Web page on which space has been purchased for dynamic advertising. In step 801, the user issues an HTTP request for such a publisher Web page.

In step 802, the facility receives the requested Web page. The received Web page contains a reference to a conditional dynamic advertisement inclusion that will determine whether it is possible to perform dynamic advertising in the client computer system, or whether, inversely, conventional banner advertising should be substituted for dynamic advertising. Table 6 below shows the HTML code preferably included in the publisher Web page for this purpose.

40	1.	<IFRAME MARGINHEIGHT=0 MARGINWIDTH=0 HEIGHT="60" WIDTH="468" SCROLLING="NO"
	2.	SRC="http://client-side component.avenuea.com/coho/cohoframe.html">
	3.	
	5.	</IFRAME>

Table 6

It can be seen that the HTML code shown in Table 6 determines whether the browser in which it is processed supports the IFRAME established by the HTML code shown in Table 6 tags. If so, the browser interprets lines 1-2 in order to retrieve the conditional dynamic advertising inclusion from the URL `http://client-side component.avenuea.com/coho/cohoframe.html`. If the browser interpreting this HTML code does not support the IFRAME tag, it interprets line 3-4 to retrieve a conventional banner advertisement from the URL `http://view.avenuea.com/view/test/direct/01`. In an alternative embodiment, the IFRAME tag is omitted from the publisher HTML code, which instead unconditionally includes the conditional dynamic advertising inclusion.

In step 803, in browsers that support the IFRAME tag, the facility retrieves the conditional dynamic advertisement inclusion using the reference contained in the requested Web page. Table 7 below shows an example of the conditional dynamic advertisement inclusion.

```

1.  <html>
20 2.  <SCRIPT LANGUAGE="JavaScript">
3.    if ( window.navigator.javaEnabled() == true )
4.    {
5.      document.write("<applet code=\"Adlet\" archive=\"Adlet.jar\"
6.      codebase=\"http://client-side component.avenuea.com/coho/\" id=\"Adlet\"
25  width=\"468\"
7.      height=\"60\">")
8.      document.write("<param name=SiteID
9.      value=\"client-side component\"></applet>")
10.   }
30 11. else
12.   { document.write("<a
13.   href=\"http://view.avenuea.com/go/test/direct/01\" target=_top> <img
14.   src=\"http://view.avenuea.com/view/test/direct/01\" width=\"468\"
15.   height=\"60\"> </a>")
35 16.   }
17. </SCRIPT>
18. </html>

```

Table 7

40

The conditional dynamic advertisement inclusion contains a JavaScript script. The script in line 3 tests whether Java applets are supported by the current browser and have been enabled. If so, lines 5-9 add to the IFRAME established in the HTML code shown in Table 6 an inclusion reference for the dynamic advertising applet. Otherwise, lines 12-15 add a reference to a traditional banner ad to the IFRAME. In additional embodiments, the conditional dynamic advertising inclusion includes additional tests regarding the suitability of the browser and/or the client computer system for supporting dynamic advertising. For example, in one such embodiment, the conditional dynamic advertisement inclusion includes tests for browser versions whose Java virtual machine implementations properly execute the dynamic advertising Java Applet.

In step 804, the facility executes the script in the dynamic advertisement inclusion to determine whether a dynamic advertisement can be presented on the client system, that is, whether the client system supports Java applets. In step 805, if a dynamic advertisement can be presented, then the facility continues in step 807, else the facility continues in step 806. In step 806, the facility retrieves and displays a conventional banner advertisement. After step 806, the steps conclude.

In step 807, the facility retrieves the dynamic advertising applet, also called the client-side component, from the dynamic advertising server. In step 808, the facility launches the retrieved applet in a separate applet thread. When it executes, the applet displays advertising messages and registers event handlers for user interface events that occur in conjunction with the dynamic advertising region. Execution of the applet is discussed in greater detail below in conjunction with Figure 9. After step 808, the facility, in the browser thread, processes user interface events. In step 809, the facility receives a user interface event. In step 810, the facility identifies an event handler registered to receive the user interface event. For user interface events occurring in conjunction with the dynamic advertising region -- that is, for those

events occurring within the boundaries of this region -- the facility has registered an event handler for recording the user interface event. In step 811, the facility invokes the event handler identified in step 810. After step 811, the facility continues in step 809 to receive and process the next user interface event.

Figure 9 is a flow diagram showing the steps preferably performed when the dynamic advertising applet is launched in a separate thread in step 808. In step 901, the facility obtains a dynamic advertising script from the dynamic advertising server. The facility preferably uses a script identifier included in the publisher Web page or other similar indicia in order to select one dynamic advertising script from a number of different dynamic advertising scripts available from the dynamic advertising server. In step 902, the facility constructs an internal representation of the script obtained in step 901. In step 903, the facility registers user interface event handlers for subregions of the dynamic advertising region specified in the script. In step 904, the facility retrieves and presents in the dynamic advertising region the initial contents specified by the script. In step 905, the facility initiates retrieval of the next advertising message contents specified by the script. In step 906, the facility sleeps for a display interval period specified by the script, which indicates the amount of time each advertising message in the sequence of advertising messages is to be displayed. In step 907, after awakening at the end of the display interval period, the facility displays the next advertising message content specified by the script, applying any visual transitions specified by the script. In step 908, if the time since the last log upload exceeds a reporting interval specifying how frequently such upload should occur, then the facility continues in step 909, else the facility continues in step 905. In step 909, the facility transmits log entries to the dynamic advertising server. In step 910, the facility deletes the transmitted log entries from the log. After step 910, the facility continues in step 905.

In additional embodiments, in addition to or in place of the test shown in step 908 for determining whether to transmit log entries to the dynamic advertising server, the facility utilizes other tests, including whether the size of the log has exceeded a particular size, such as the size of a packet used to transmit the log entries, or a multiple thereof; whether the log contains a particular type of entry, such as a mouse-click entry in a particular region; or any combination of these or other factors.

Figure 10 is a flow diagram showing the steps preferably performed in an event handler registered by the facility to process user input events occurring in conjunction with the dynamic advertising region. In step 1001, the facility adds to the event log an entry describing the current user interface event. Log entry formatting is described in greater detail below in conjunctions with Tables 8 and 9. In step 1002, the facility processes the event if any processing is required for the event. For example, for mouse-click events, the facility preferably identifies the subregion containing the coordinates of the click, and causes the browser to navigate to a URL specified by the script for that subregion. After step 1002, the steps conclude.

In one embodiment, the event handlers provided by the facility write four different types of log entries, which are described below in Table 8.

25	Entry Type 1 - Impression Request ScriptID <tab> Sequence # <tab> Cookie <tab> Entry Type <tab> Time of Impression ¹
30	Entry Type 2 - Image Display Time ScriptID <tab> Sequence # <tab> Cookie <tab> Entry Type <tab> Start Time ¹ <tab> Stop Time ¹ <tab> Image Number from script
35	Entry Type 3 - Click ScriptID <tab> Sequence # <tab> Cookie <tab> Entry Type <tab> Time of Click ¹ <tab> Image Number from script <tab> Region number within image <tab> Click-through URL
	Entry Type 4 - Mouse Over Image Time ScriptID <tab> Sequence # <tab> Cookie <tab> Entry Type <tab> Start Time ¹ <tab> Stop Time ¹ <tab> Image Number from script <tab> Region number within image

¹ # of milliseconds since 1/1/1970

Table 8

5 A sample log corresponding to the example is shown below in Table 9.

1. {3595209E-1558-11D4-934F-00C04F08320A} 1	949013076-280132	1	956090465522			
2. {3595209E-1558-11D4-934F-00C04F08320A} 2	949013076-280132	2	956090465662	956090467745	0	
3. {3595209E-1558-11D4-934F-00C04F08320A} 3	949013076-280132	2	956090467745	956090470629	1	
4. {3595209E-1558-11D4-934F-00C04F08320A} 4	949013076-280132	2	956090470629	956090477649	0	
5. {3595209E-1558-11D4-934F-00C04F08320A} 5	949013076-280132	4	956090480233	956090480603	1	1
6. {3595209E-1558-11D4-934F-00C04F08320A} 6	949013076-280132	4	956090480604	956090481595	1	0
7. {3595209E-1558-11D4-934F-00C04F08320A} 7	949013076-280132	2	956090477649	956090481599	1	
8. {3595209E-1558-11D4-934F-00C04F08320A} 8	949013076-280132	3	956090481600	1	0	http://www.avenuea.com

Table 9

10

Log entry 1 indicates the retrieval of the binary script from the dynamic advertising server. Log entries 2, 3, 4, and 7 indicate the display of advertising messages. Log entry 8 indicates a click within an advertising message. Log entries 5 and 6 indicate that the user has
 15 "moused-over" an advertising message.

It will be understood by those skilled in the art that the above-described facility could be adapted or extended in various ways. For example, various types of client-side components could be used other than Java applets. Also, the facility may collect a variety of different
 20 types of information relating to the user's viewing of an interaction with advertising messages or incident thereto. Further, the dynamic advertising server may do a variety of types of analysis on the data returned by the client-side component, and may, based on such analysis, adjust the performance of dynamic advertising by the client-side component, either
 25 by providing a revised script or by issuing other types of instructions. Also, the facility may present a wide variety of types of advertising messages, including static images; video sequences; animation sequences

such as animated GIFs, SHOCKWAVE™ animation sequences, FLASH™ animation sequences; streaming sequences; and Java-implemented advertising messages, either homogeneously or heterogeneously. While various embodiments of the client-side
5 component that execute in conjunction with a Web browser on a general-purpose computer system are discussed above, various additional embodiments of the client-side component execute in different environments, including such electronically-addressable special-purpose devices as cellular phones, pagers, personal digital assistants, automatic
10 teller machines, televisions and devices connected thereto, and other home and auto appliances, as well as in various additional environments. While the foregoing description makes reference to preferred embodiments, the scope of the invention is defined solely by the claims that follow and the elements recited therein.

CLAIMS

1 1. A method in a client computer system for presenting
2 dynamic advertising, comprising:
3 receiving a specified sequence for advertising messages that
4 identifies advertising messages by reference;
5 retrieving advertising messages identified in the specified
6 sequence;
7 displaying advertising messages in accordance with the
8 specified sequence;
9 collecting information describing interaction by a user of
10 the client computer system with the displayed advertising messages; and
11 transmitting the collected information to a separate
12 information processing computer system.

1 2. The method of claim 1 wherein one of the
2 advertising messages specified in the sequence is a static image.

1 3. The method of claim 1 wherein one of the
2 advertising messages specified in the sequence is an animation sequence.

1 4. The method of claim 1 wherein one of the
2 advertising messages specified in the sequence is a video sequence.

1 5. The method of claim 1 wherein one of the
2 advertising messages specified in the sequence is a rich media element.

1 6. The method of claim 1, further comprising:
2 receiving a second specified sequence for advertising
3 messages that identifies advertising messages by reference, wherein at
4 least one of the advertising messages identified by the second specified
5 sequence is identified by the specified sequence and has therefore been
6 retrieved; and
7 displaying advertising messages in accordance with the
8 second received sequence without re-retrieving the advertising messages
9 identified in both the second specified sequence and the specified
10 sequence.

1 7. The method of claim 1, further comprising
2 determining that a predetermined period of time has passed since
3 collected information was last transmitted to the separate information
4 processing computer system, and wherein the transmitting is performed in
5 response to the determination.

1 8. The method of claim 1, further comprising
2 determining that the collected information has reached a predetermined
3 size, and wherein the transmitting is performed in response to the
4 determination.

1 9. The method of claim 1 further comprising
2 determining that the collected information contains information describing
3 a predetermined type of user interaction, and wherein the transmitting is
4 performed in response to the determination.

1 10. The method of claim 1 wherein the transmitting is
2 performed in response to the first of the following:

3 determining that a predetermined period of time has passed
4 since collected information was last transmitted to the separate
5 information processing computer system;

6 determining that the collected information has reached a
7 predetermined size; and

8 determining that the collected information contains
9 information describing a predetermined type of user interaction.

1 11. The method of claim 1, further comprising, in the
2 information processing computer system:

3 receiving the collected information transmitted from the
4 client computer system; and

5 analyzing the received information to discern the effect of
6 the specified sequence of advertising messages on the user.

1 12. The method of claim 11 wherein the results of the
2 analysis characterize the user.

1 13. The method of claim 11 wherein the results of the
2 analysis characterize the sequence of advertising messages.

1 14. The method of claim 11 wherein the results of the
2 analysis characterize one or more of the advertising messages of the
3 sequence.

1 15. The method of claim 1, further comprising displaying
2 a visual transition between the display of two advertising messages that
3 are adjacent in the sequence.

1 16. The method of claim 15, further comprising
2 receiving with the specified sequence a specified transition time, and
3 wherein the visual transition is displayed for the specified transition time.

1 17. The method of claim 15, further comprising:
2 during the display of the visual transition, detecting a
3 selected user interaction performed with respect to the displayed visual
4 transition;
5 in response to so detecting:
6 interrupting the display of the visual transition; and
7 resuming the display of the first of the adjacent advertising
8 messages.

1 18. The method of claim 17 wherein the selected user
2 interaction is moving a mouse cursor into the region in which the visual
3 transition is being displayed.

1 19. The method of claim 1 wherein each of the displayed
2 advertising messages is displayed on behalf of a different advertiser.

1 20. The method of claim 1 wherein at least two of the
2 advertising messages are displayed on behalf of different advertisers.

1 21. The method of claim 1 wherein each of the displayed
2 advertising messages is retrieved from a different server.

1 22. The method of claim 1 wherein the displaying
2 displays multiple advertising messages simultaneously.

1 23. The method of claim 1, further comprising requesting
2 the received sequence from among a plurality of available sequences.

1 24. The method of claim 23 wherein retrieval of the
2 sequence is triggered by processing a publisher Web page, and wherein
3 the publisher Web page includes an identifier, the method further
4 comprising selecting the requested sequence based upon the identifier.

1 25. The method of claim 1 wherein each advertising
2 message having a predecessor in the sequence is retrieved during the
3 display of its predecessor.

1 26. The method of claim 1 wherein both a first and a
2 second sequence for advertising messages are received, both the first
3 sequence and the second sequence identifying a first advertising message,
4 and wherein advertising messages are displayed in accordance with both
5 the first and second sequences, and wherein the first advertising message
6 is retrieved only once.

1 27. A computer system for presenting advertising,
2 comprising:
3 a first receiver that receives an advertising message;
4 a display that displays the received advertising message in
5 an advertising region;
6 a second receiver that receives a specification specifying a
7 plurality of subregions of the advertising region within which to monitor
8 user interactions;
9 a user interaction monitoring subsystem that, during the
10 display of the advertising message, monitors user interactions within the
11 specified subregions of the advertising regions;

12 a user interaction logging subsystem that logs monitored
13 user interactions; and
14 a transmitter that periodically transmits the logged user
15 interactions to a separate information processing computer system.

1 28. A computer memory containing a dynamic
2 advertising specification data structure for use by a client-side advertising
3 agent, comprising:
4 a first portion indicating a display sequence of advertising
5 messages;
6 a second portion indicating one or more regions within the
7 advertising messages of the indicated display sequence in which to
8 monitor user interactions,
9 such that, when the data structure is in the possession of a client-side
10 advertising agent, the advertising agent may access the first portion in
11 order to display the display sequence of advertising messages, and may
12 access the second portion to identify regions within the advertising
13 messages in which to monitor user interactions.

1 29. The computer memory of claim 28 wherein the data
2 structure is expressed in XML.

1 30. The computer memory of claim 28 wherein the data
2 structure is expressed in a compressed form of XML.

1 31. A computer-readable medium whose contents cause
2 a user computer system to advertise, comprising:
3 retrieving advertising instructions;
4 in accordance with the advertising instructions, retrieving
5 advertising content;

6 composing an advertising message incorporating the
7 advertising content;
8 presenting the composed advertising message;
9 monitoring user behavior in response to the presentation of
10 the composed advertising message; and
11 periodically transmitting an indication of the monitored user
12 behavior to a destination computer system distinct from the user computer
13 system.

1 32. The computer-readable medium of claim 31 wherein
2 the advertising instructions include a specification specifying a plurality
3 of subregions of the advertising region within which to monitor user
4 interactions,
5 and wherein the monitoring monitors user interactions
6 within the specified subregions of the advertising regions.

1 33. The computer-readable medium of claim 32 wherein
2 the received specification further specifies for a selected one of the
3 specified regions a URL to navigate to if the user clicks in the selected
4 region, the contents of the computer-readable medium further causing the
5 user computer system to navigate to the URL specified for the selected
6 region when the user clicks in the selected region.

1 34. The computer-readable medium of claim 32 wherein
2 the received specification further specifies text to display in a selected one
3 of the specified regions, the contents of the computer-readable media
4 further causing the user computer system to display the specified display
5 text within the selected region.

1 35. The computer-readable medium of claim 32 wherein
2 the received specification further specifies for a selected one of the

3 specified regions audio to play if the user moves a mouse cursor into the
4 selected region, the contents of the computer-readable media further
5 causing the user computer system to play the audio specified for the
6 selected region when the user moves the mouse cursor into the selected
7 region.

1 36. The computer-readable medium of claim 32 wherein
2 the received specification further specifies for a selected one of the
3 specified regions cursor text to display in conjunction with a mouse
4 cursor if the user moves the mouse cursor into the selected region, the
5 contents of the computer-readable media further causing the user
6 computer system to display the specified cursor text in conjunction with a
7 mouse cursor when the user moves the mouse cursor into the selected
8 region.

1 37. The computer-readable medium of claim 32 wherein
2 the received specification further specifies for a selected one of the
3 specified regions an alternative cursor to substitute for a mouse cursor if
4 the user moves the mouse cursor into the selected region, the contents of
5 the computer-readable media further causing the user computer system to
6 substitute the specified alternative cursor for the mouse cursor when the
7 user moves the mouse cursor into the selected region.

1 38. A generated data signal conveying an advertising
2 data structure, comprising:
3 information specifying an advertising message;
4 information defining a first region of the advertising
5 message, the defined first region having a first property value of a first
6 property type; and

7 information defining a second region of the advertising
8 message, the defined second region having a second property value of a
9 second property type,
10 such that, when the data signal is received in a computer system, the
11 computer system may use the contents of the data structure to render the
12 specified advertising message with the defined regions.

1 39. The data signal of claim 38 wherein the first and
2 second regions are coextensive.

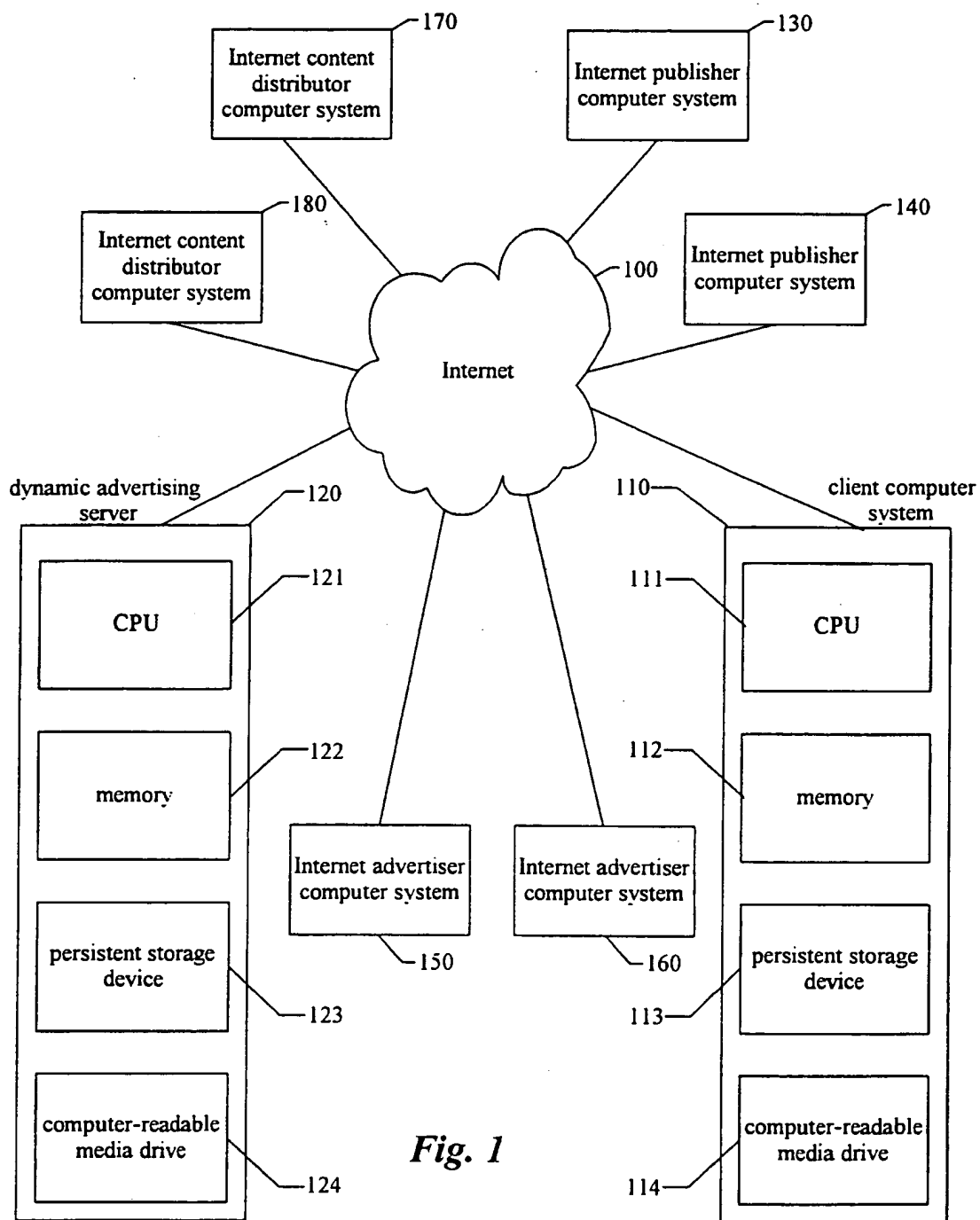
1 40. The data signal of claim 39 wherein the first and
2 second property types are distinct.

1 41. The data signal of claim 38 wherein the first and
2 second property types are the same, and wherein the first and second
3 property values are distinct.

1 42. The data signal of claim 38 wherein the first and
2 second regions are noncoextensive.

1 43. The data signal of claim 38 wherein the first property
2 type is click-through, and wherein the data structure further comprises
3 information specifying a click-through target address to which to navigate
4 when the user clicks within the first region.

1 44. The data signal of claim 38 wherein the first property
2 type is superimposed text, and wherein the data structure further
3 comprises information specifying text that is to be superimposed within
4 the first region.



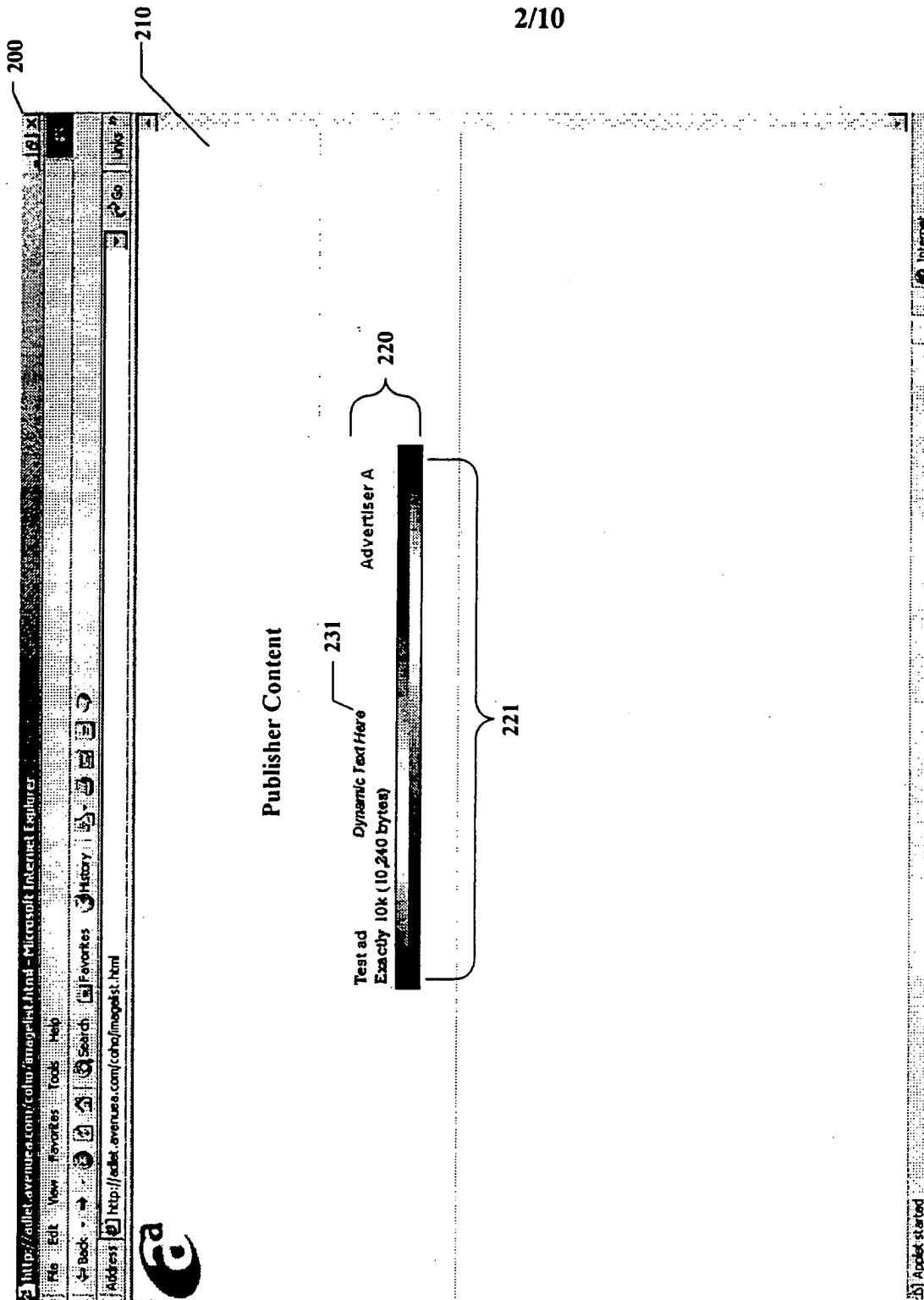


Fig. 2

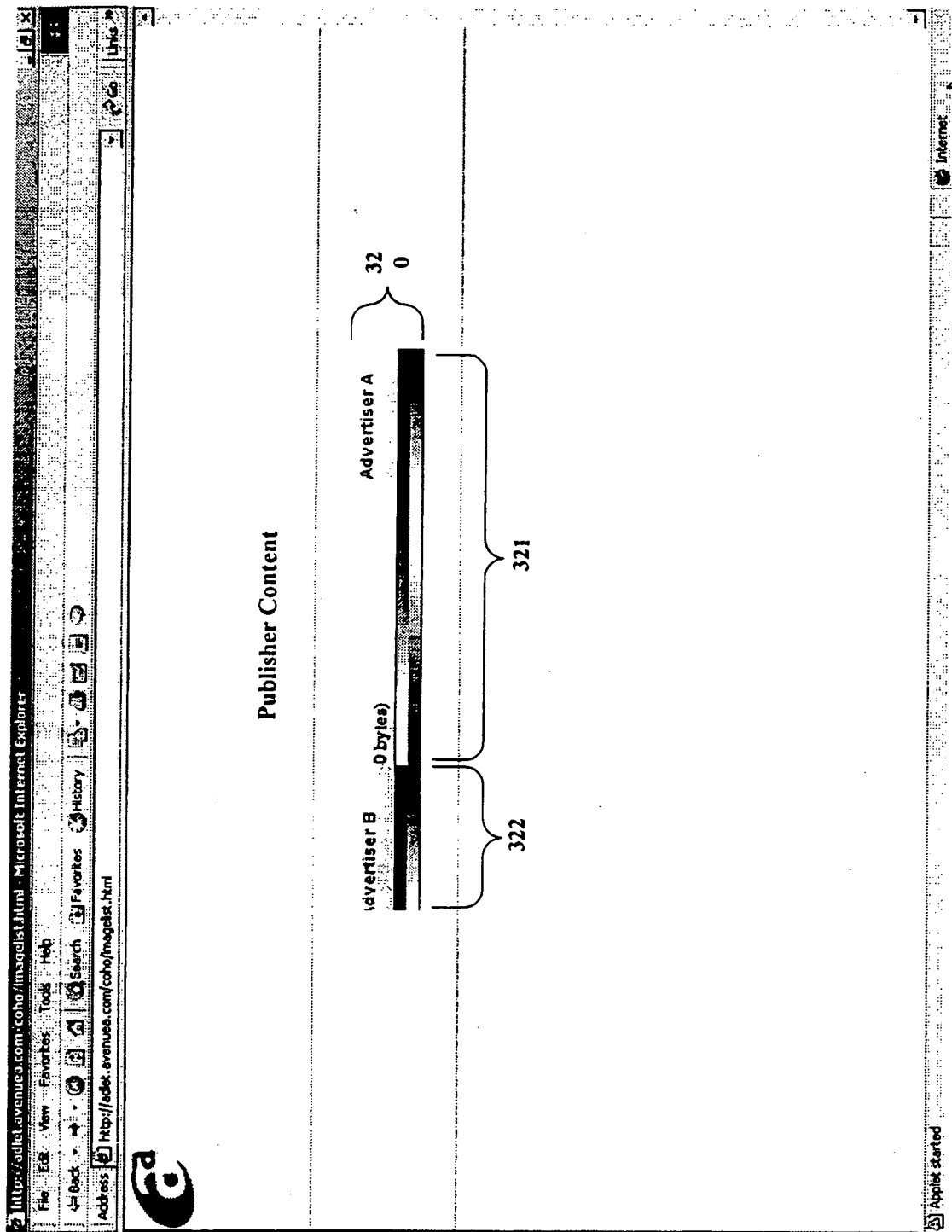


Fig. 3

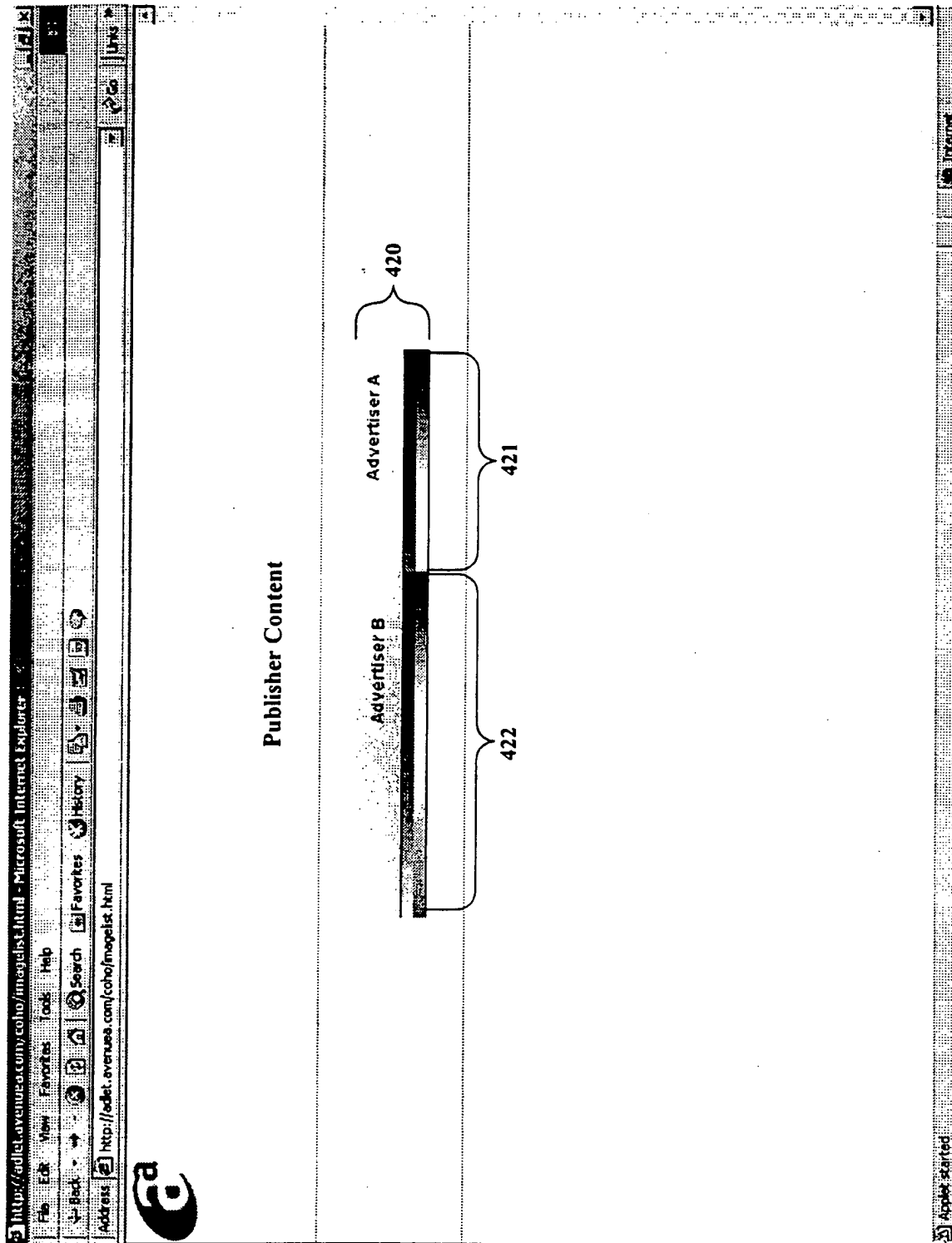


Fig. 4

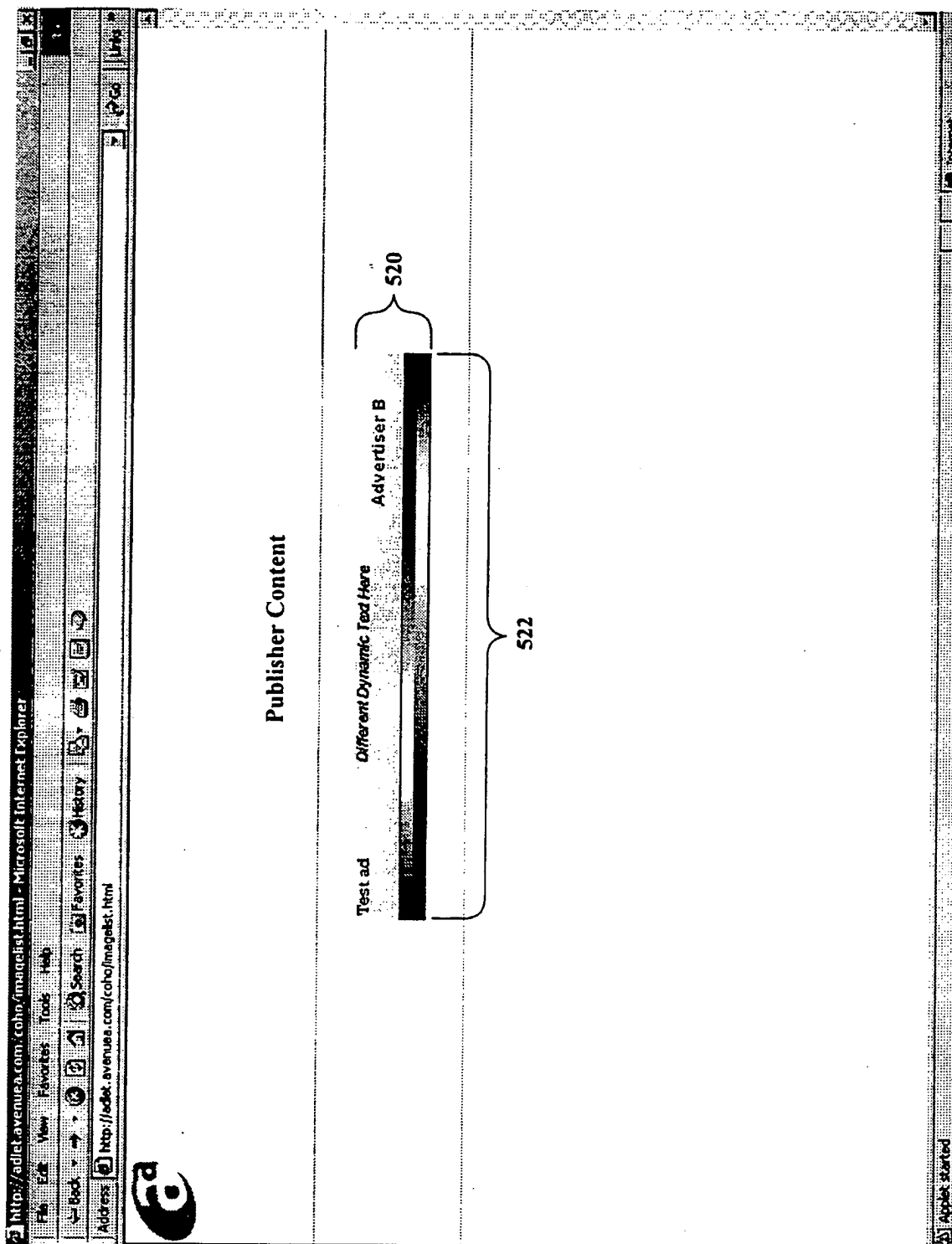


Fig. 5

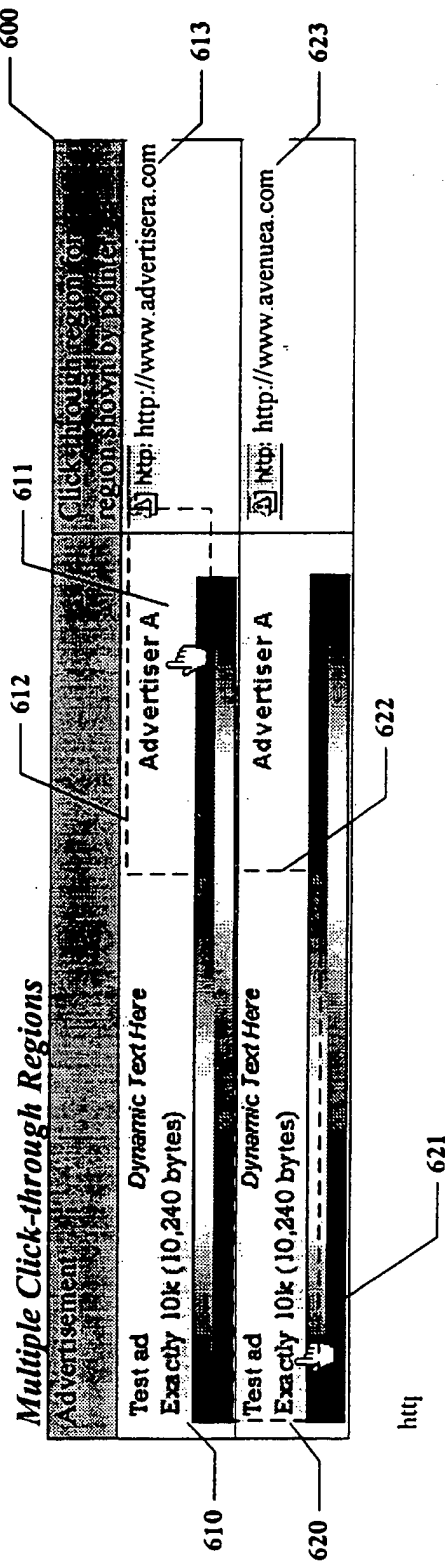


Fig. 6

7/10

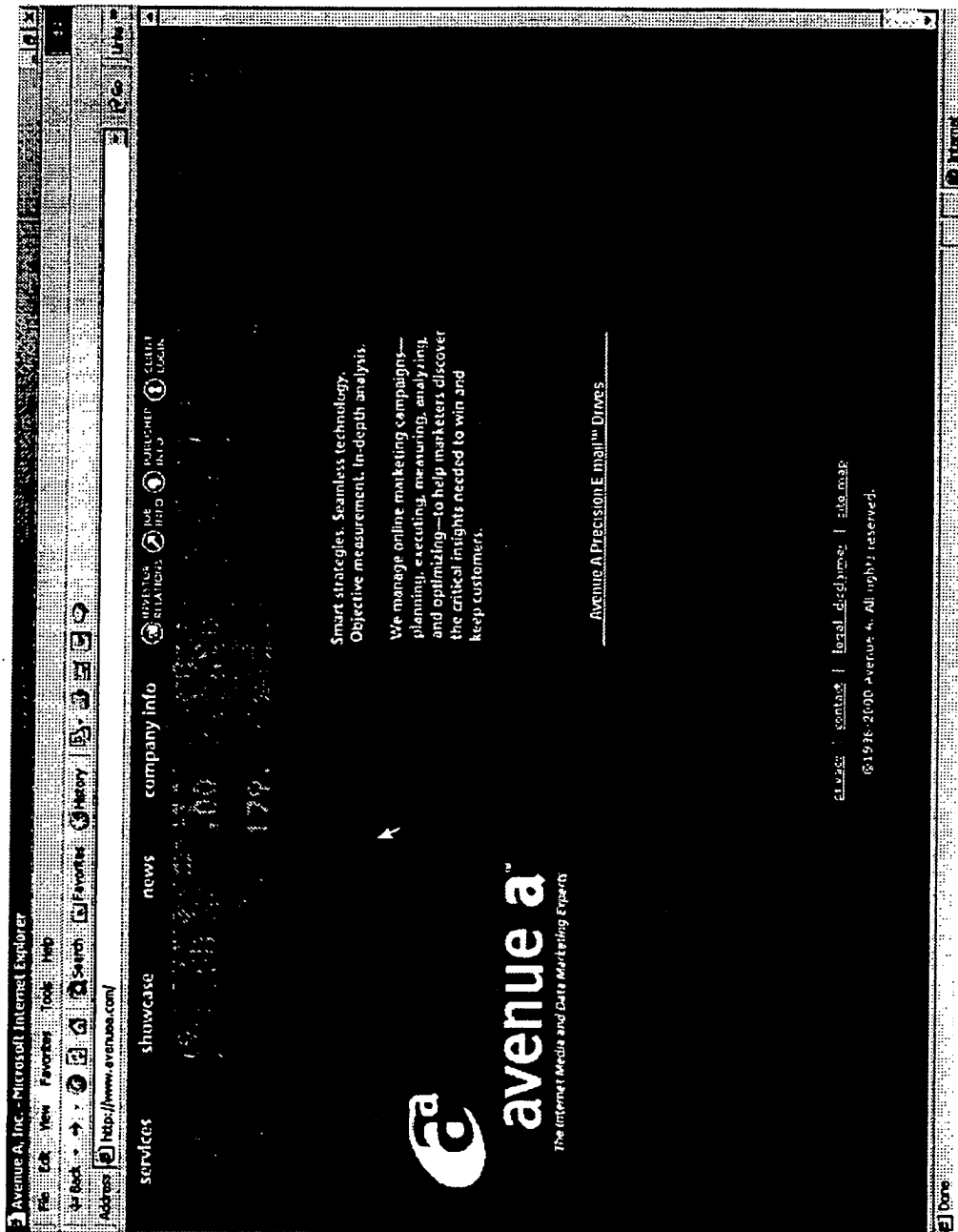
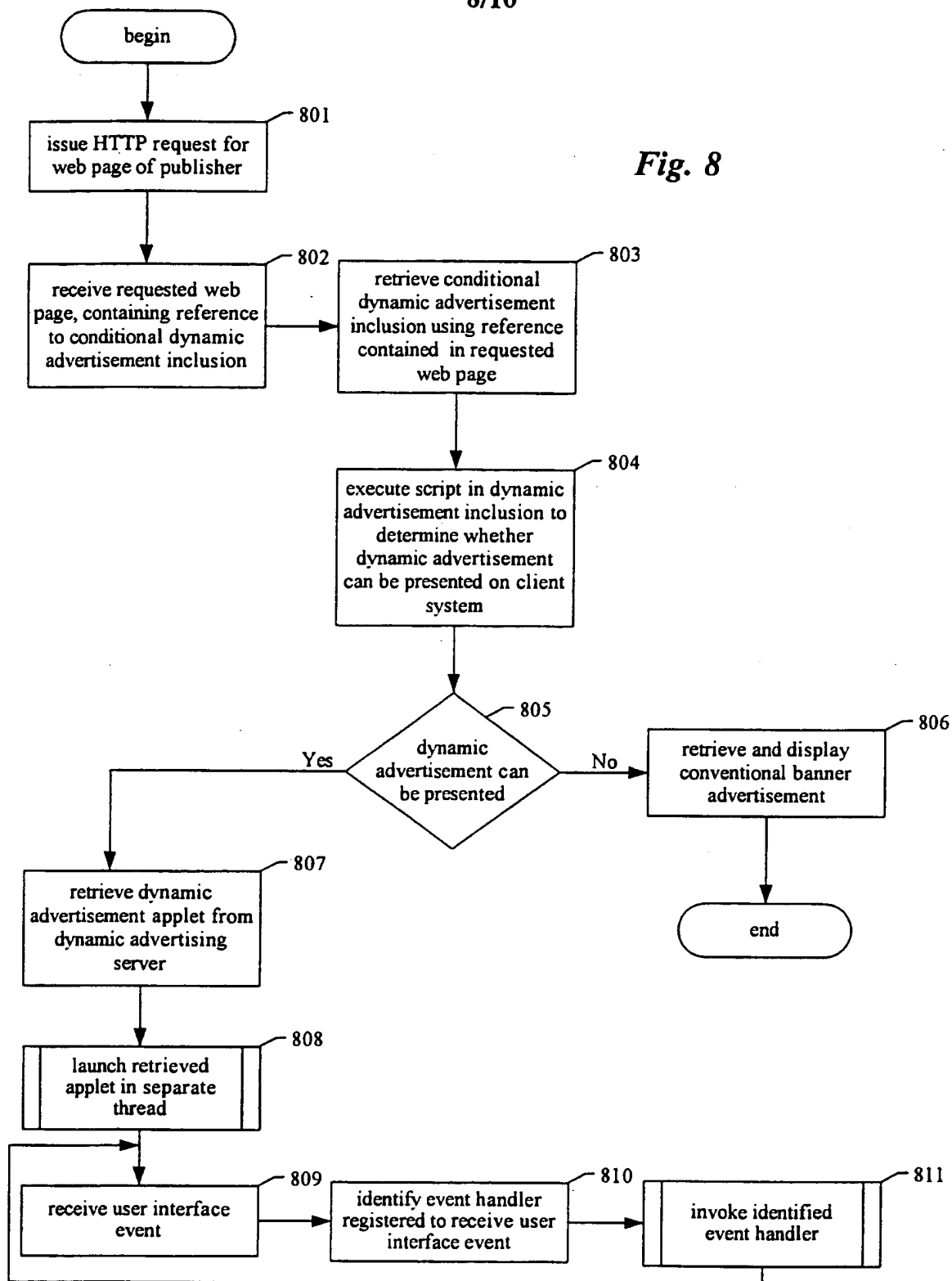
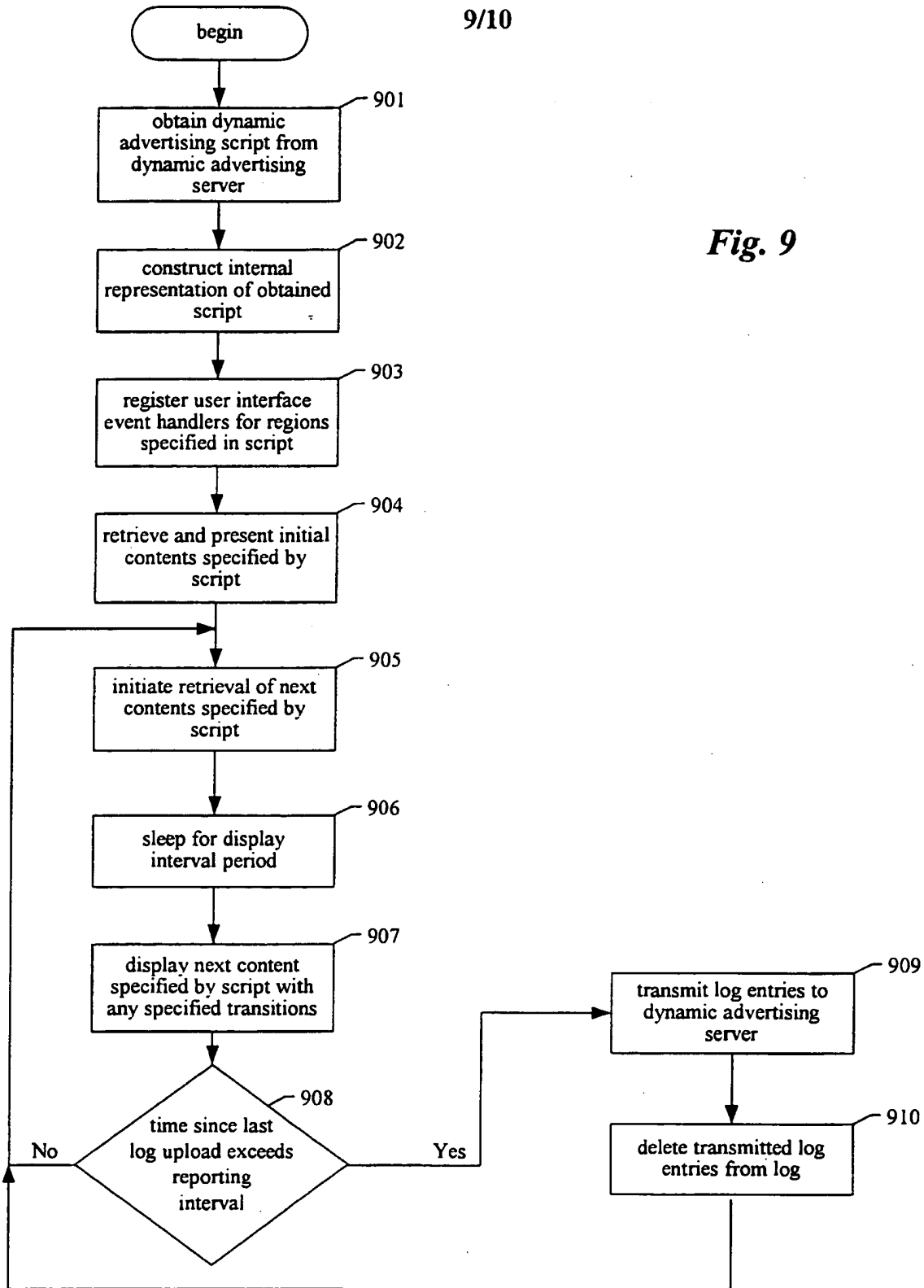


Fig. 7

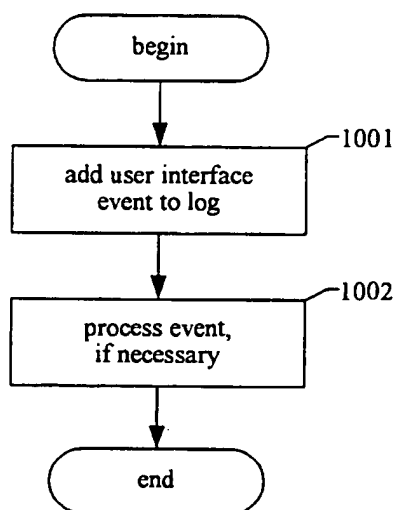
8/10

Fig. 8

9/10

Fig. 9

10/10

*Fig. 10*